Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 1 Issue 4, November - 2014

A Level Set Approach for High Performance Computing for Multiphase Fluid Flows

A K Borah
Department of Mathematics, R G Baruah College, Gauhati University, Pin -781025 India E -Mail : borah.arup@yahoo.com

P K Singh
Department of Mathematics, University of Allahabad, Uttar Pradesh, Pin -211002, India E -Mail: pramod_ksingh@rediffmail.com

Keywords - Fluid flow modelling, Krylov Subspace, volume of fluid (VOF), N-S solver, interface tracking method, colour function.

***Abstract*--** The study of fluid dynamics has applications in many areas of engineering and science. This includes the understanding environmental, biological and chemical flows. Laboratory experiments and computer simulations are the two main approaches for the study of complex flow problems. Computer simulations can help one to understand and analyse the complexity involved in these flows more clearly without having to perform time consuming, expensive and complicated experiments. Advancements in modern computers have facilitated the solution and analysis of fluid flows with high accuracy (close-to-reality) and reduced computational time. In computational fluid dynamics, one has to employ basic physical principles to develop mathematical models and therefore obtain accurate numerical solutions. Development and improvements of numerical schemes have encouraged researchers to investigate almost too every branch of fluid dynamics and its applications to real life problem. But multiphase flows occur in many industrial and natural phenomena such as petroleum refining, biological flows and interaction with air with sea surface. The simulation of multiphase fluid flows is one of the most challenging problems in CFD as it involves the modelling of sharp interfaces separating multiple fluids.

I. INTRODUCTION

The physical behaviour of two-phase flows can be predicted by computer modelling [1]. Recent advances in computational methods for multiphase flow involving gas, solid and liquid phases [2-3]. The mathematical interpretations of the mass and momentum conservation principles [4]. The governing equations are presented by a set of PDEs [5].

The continuity equation at a point in a compressible fluid such as

$$\frac{\partial}{\partial t}(\rho) + \nabla.(\rho V) = 0 \qquad (1)$$

Where t, ρ, V and ∇ denotes time, density, velocity vector and nabla operator. The transportation of fluid is described by the momentum equation as

$$\left(\frac{\partial}{\partial t}(\rho V) + \nabla.(\rho V V)\right) = -\nabla \mathrm{p} + \gamma \nabla^2 V + S_v \qquad (2)$$

Equations (1)-(2) are Navier Stokes equation [6], which are difficult to solve analytically and numerical solutions are applied for simulation. Furthermore, in the simulation the properties of the fluid change abruptly across the interface whose location must be known at every time-step. This position is determined by interface advection methods and fluid motion is predicted applying N-S solvers [7-8]. The main two approaches to the solution of the multiphase fluid flows- air and water flow which involves large viscosity and density ratios. On the other hand, in the first approach only the liquid phase (to be inviscid) is modelled. But N-S equations can be converted into Laplace's equation [9]. In

the second approach viscous effects have been considered [1]

And the both the liquid and gas are modelled with the full N-S equations.

We mainly focussed on the second approach for multiphase flow the flow of two-immiscible fluids (multifluid flow).

2. *Numerical Methods* The numerical solution calls for the discretization of the computational domain using a grid arrangement to store variables for the fluid properties. Therefore, the PDEs are discretized according to the chosen grid resulting in a linear system of equations which is then solved to obtain the numerical solution.

2.1 *Method for multiphase flows*- Modelling the advection of the sharp interface numerically is a challenging task [10] and range of methods [11] have been developed over the last 30 years to achieve this. We studied two phase flows applied to immiscible fluids can be classified into two categories – *interface tracking* and *interface capturing* [7][12-13].

2.1.1 *The interface tracking method*- the interface between two fluids is explicitly tracked during the fluid motion to maintain a sharp discontinuity in the fluid viscosity and density. The motion of the interface is tracked either by marker points located on the surface or by attaching it to a mesh boundary surface.

The second is the front tracking method, a separate front, on a fixed grid is used to identify the interface for each phase with the help of additional computational elements introduced explicitly. These elements are marker particles on the surface and form a moving internal boundary. Both phases are treated are one fluid with variable materials properties and one set of the N-S equations is solved over the whole computational domain. Furthermore, fluid properties, density and viscosity are calculated with respect to the interface position. On the other hand, this method uses piecewise linear, higher order polynomial to fit the interface which is advected with the flow fields in a Lagragian manner [14-15]. Henceforth another interface tracking method is the moving mesh method [16-17] wherein a boundary fitted grid is employed and the grid points are embedded in the fluid and these points are moved in a Lagragian manner. The same fluid elements are kept in all computational cells which are adjacent to the interface and the fluid always coincides with the specified regions facilitating the piecewise tracking of the fluid surface [18]. The interface is integrated with these points and is tracked by the nodes affixed on both phases. The movement of the interface is determined utilizing the knowledge of velocities known at the current time-step [19].

Moreover, N-S equations are solved for both fluids (liquid and gas). For the liquid phase, these equations are solved on a deforming unstructured mesh [17] and one of the advantages of this method is that it permits the accurate prescription of the boundary conditions of the interface [20].

2.1.2 *The interface capturing method* - The volume fraction function which as the colour function is a step function and represents the fraction of volume occupied by one fluid. The interface is reconstructed from the value of the colour function [21-22]. The volume fraction method solves a scalar transport equation Eq.(3) in an Eulerian manner

$$\frac{\partial}{\partial t}(C) + V.\nabla \mathrm{C} = 0 \qquad (3)$$

where C being the colour function is a step-function and represents the fraction volume occupied by one fluid shown in

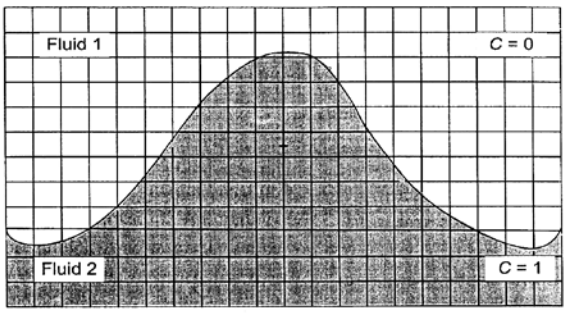

Fig. 1 Interface capturing methods

which shows that the value of the colour function C is only in the fluid 2 and zero in the fluid 1 unity while it lies between zero and unity at the interface. The mesh is kept fixed and a suitable technique is chosen to locate the interface in the interface capturing methods [14][23]. Since the interface is

reconstructed at each time-step from the known value of the indicator function and also this method is able to cope with large stretching and deformation of the interface. On the other hand, this methods can be used for modelling large scale deformations of the interface including breakup and merging [7]. The volume fraction method which uses the colour functions to identify the interface. The volume of fluid methods has been widely used for the numerical simulation of viscous flows having non-zero viscosity with moving interface.

II. KRYLOV-SUBSPACE METHOD (K-S)

Krylov–Subspace method [24-25] are some of the most efficient iterative methods applied to a large sparse linear system of equations obtained from PDEs representing multiphase flow problems [15]. A residual vector $\gamma = B - Ax$ is calculated initially and is minimized or is set to satisfy some other constraints in each iteration. The residual vector on the multiplication by powers of the matrix A in successive iteration, generates a subspace which is Krylov Subspace (K-S) [26-27], as *m* denotes dimension of the subspace. This method belong to non stationary category [28] as shown in Fig.2

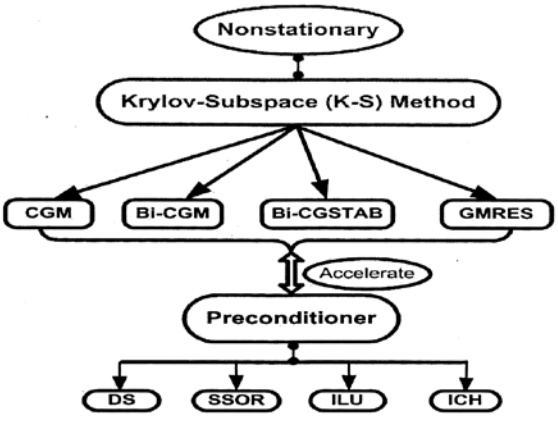

Fig. 2 Nonstationary iterative methods

have been widely adopted as a good choice for solving large sparse linear systems of equations [29-30]

III. MODEL VALIDATIONS AND OTHER NUMERICAL SCHEME

The whole model for multiphase flow problems consists of two parts are shown in Fig.3

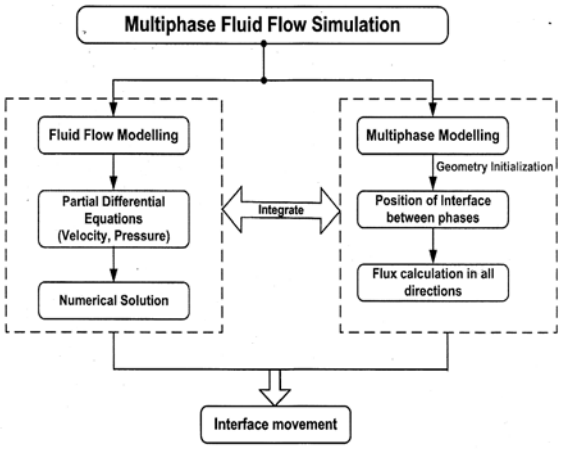

Fig.3 Different steps of the multiphase fluid flow simulation

and these parts are PDEs solvers which includes linear system solvers and multifluid methods [31-32]. Furthermore, in case of computer simulation both two parts are integrated together and executed within a time *iteration* [33].

In order to validate this integrated model many benchmark problems [34-35].

4.1 *Parallel Numerical Algorithm* – Computer simulation of multiphase flow problems involve the implementation of complex algorithms which are as intensive in both time and memory requirements [36-37]. Executing the Algorithm may take days when run on a single processor. Moreover, in the this studies simulation of some problems – dam breaking , the rising of a air bubble, and the fine grid resolution 512×512 takes 6-7 days when executed in a single processor with clock speed 2.8GHz and 8 GBRAM. Henceforth, for higher resolution simulation of CFD problems the matrix size may become too big to fit in the memory and thus, it is difficult to solve the linear system on the available computer [38-39].

To deal with memory constraint and to reduce computational time, these complex algorithm need to be implemented on parallel computers consisting the clusters of processors [40-41]. An idea of the importance of the parallel computers can be found from the top 500 site (www.top500.org) the five hundred fastest machines in the world on their performance on the various benchmarks problems. In this

approach one can avail of the huge memory and computational power of many processors. On the other hand, the parallel implementation of CFD code has been carried out for the simulation [42]. The development of parallel algorithms has been focus of intense research in the area of parallel matrix computation [40][43-44]. The quality of parallel algorithms can be measured by using common performance evaluators, speed up S_n and Efficiency E_n [45-46]

$$\frac{T_s}{T_p} = S_n \;\; ; \;\; \frac{S_n}{n} = E_n \qquad (4)$$

Where T_p and T_s denotes time taken by parallel algorithm on n processors and T_s denotes the time taken by its serial version. In the present work, the simulation has been carried out on a Linux cluster which has 56 nodes, each having 2 quad core processors with clock speed 2.8GHz and 8 GBRAM. In the programs related to parallel matrix computation, processor is required to send the executed data to each other within a network [48]. Now, these data are sent in the form of message by using certain parallel communication libraries Message Passing Interface [31] [49-51]. Now for parallel computation of the linear solvers one processor is required to manage the processor communication such as gathering the parts of matrix-vector products calculated by other processor and therefore [43] and these processor act the master processor and therefore the master slave paradigm [42][51] has been adopted in the research studies to design the parallel algorithms for the different steps of the linear solvers

.

DISCUSSION AND FURTHER RESEARCH

The following conclusions can be drawn from the above studies:

[i] computer simulation of multiphase flow requires the numerical solution of PDEs. The numerical solution is used to move the interface between two-phases of fluids in the multifluid model.

[ii] the interface capturing method is more suitable for large deformation and stretching on the interface.

[iii] the VOF (PLIC) method maintains the sharpness of the interface.

[iv] an analytic relation between the *interface position* and the volume fraction facilitates extension to 3D problems.

[v] PDEs are solved numerically in a computational domain which is divided into parts using suitable grid schemes. *N*ow, the under mentioned points and observations are noted for steps from domain discretization to linear system of equations:

[a] the FVM enforces conservation the momentum and mass in each CV as well as in the whole domain.

[b] the staggered grid is more suitable to deal with the problem involving velocity–pressure coupling.

[c] the non-linearity of velocity terms can be dealt with by the SIMPLE method.

[d] a special data structure is required to store sparse matrices.

[e] the non-stationary K-S solver Bi-CGSTAB provides smooth convergence.

[f] effective preconditioners for K-S solvers are DS, ILUT and SSOR.

[g] large scale problems are solved on the parallel computer.

[h] parallel algorithms are needed to develop to carry out the simulation on these computers.

ACKNOWLEDGMENT

The author would like to acknowledge the financial support from the University Grants Commission, Bahadurshah Zafar Marg, New Delhi, India, under F. No 40-236/2011 (SR).

REFERENCES

[1] A. K. Singhal, M. M. Athavale, H. Li, and Y. Jiang, “Mathematical Basis and Validation of the Full Cavitation Model,” Journal of Fluids Engineering vol.124, pp. 617–624, 2002.

[2] E. Loth, "Numerical approaches for motion of dispersed particles, droplets and bubble," Progress in Energy and Combustion, vol.26, pp. 161-223, 2000.

[3] B. Moses, "Simulation of the Multiphase Fluid Flow Using a Spatial Filtering Process, Ph. D Thesis Stanford University, 2007

[4] J. D. Anderson, "Computational Fluid Dynamics (The Basic with Application) McGraw- Hill Book. Co,Singapore 1995.

[5] H. K. Versteeg and W. Malalasekera, An Introduction to Complete Fluid Dynamics (The Finite Volume Approach) Pearson Prentice Hall

[6] G. Galdi, "An Introduction the Navier-Stokes Initial-Boundary Value problem," In Fundamental Direction of Mathematical Fluid Mechanics , pp 1-98, Swizerland Verlag, 2000.

[7] G. Chen and C. Kahrif, "Two-dimensional Navier-Stokes Simulation of Breaking Waves," Physics of Fluids, vol. 11(1), pp. 121–133, 1999.

[8] M. Henrie and G. Stanley, "Two-Phase Flow in Piper line Simulation Slack Line Condition in Trans-Alaska Crude Oil Pipeline," 1997, http://www.psig.org/papers/1990/9708.pdf

[9] Y. Delaure and A. Lewis, " 3D Hydrodynamic Modelling of Fixed Oscillating Water Column Wave Power Plant by Boundary Element method," Ocean Engineering, vol. 30(3), pp. 309–330.

[10] Y. Delaure, "A Hydrodynamics Investigation of Oscillating water Column Wave Energy Converters," Ph. D Thesis, 2001, University College, Cork Ireland.

[11] G. Tryggvason, B. Burner, A. Esmaeeli, D. Juric and N. Al-Rawahi, " A Front-tracking Method for the Computation of Multiphase Flow," Journal of Computational Physics, vol. 169, pp.708–759, 2001.

[12] J. Ferziger and M. Peric, "Computational Methods of Fluid Dynamics," 1999, Springer Verlag, Berlin Heidelberg, Germany.

[13] P. Hogg, X. Gu, and D. Emerson, "An Implicit Algorithm for Capturing Sharp Fluid Interfaces in the Volume Fluid Advection Method," Technical report, Council of the Central Laboratory of the Research Council (CCLRC) Daresbury Laboratory, Warrington, United Kingdom.

[14] D. Greaves, "A Quadtree Adaptive method for Simulating Fluid Flow in moving Interfaces," Journal of Computational Physics, vol. 194. pp.35–36, 2004.

[15] J. Hua, J. Stene, and P. Lin, "Numerical Simulation of Bubble Rising in Viscous Liquid Using a Front Tracking Method," Journal of Computational Physics, vol. 227, pp. 3358–3382, 2008.

[16] M. J. Hyman, "Numerical Methods for Tracking Interfaces," Physica, vol.12(D), pp. 396–407, 2008.

[17] W. Welch, "Local Simulation of Two-Phase Flows Including Interface tracking with mass Transfer," Journal of Computational Physics, vol. 121(1), pp. 142–154, 1995.

[18] M. Kim and W. Lee, "A View of VOF based Numerical Scheme for the Simulation of Fluid Flow with Free Surface: Part I: New free-surface tracking algorithm and its verification," International Journal of Numerical Fluids, vol.30, pp. 130–143, 2003.

[19] T. Tao,. " Moving Mesh Methods for Computational Fluid Dynamics," In recent Advances in Adaptive Computations Contemporary Mathematics, pp. 1–31, American Mathematical Society, 2005.

[20] O.Ubbink," Numerical prediction of two-fluid systems with sharp interfaces," Ph.D thesis University of London, 1997.

[21] C. Hrt,. and B.D. Nichols, "Methods for the Dynamics of Free Boundaries," Journal of Computational Physics, vol. 39, pp.201,1981.

[22] S. Ruben ands. Zaleski, "Interface Reconstruction with Least square Fit and Split Eulerian Lagragian Advection," International Journal of Numerical Methods in Fluids, vol.41, pp.251–274, 2003.

[23] R. Scardovellie,. and S. Zaleski, "Analytical Relations Connecting Linear Interfaces and Volume Fractions in Rectangular Grids," Journal of Computational Physics, vol.164(6), pp.228–237, 2000.

[24] O. Axelsson, "Iterative Solution Methods," Cambridge University Press, Cambridge United Kingdom, 1996.

[25] Y. Saad, "Krylov-Subspace Methods for Super Computers," SIAM Journal of Statistical and Computing, vol.10, pp.1200–1232, 1989.

[26] Y. Saad, "Iterative methods for Sparse Linear Systems," PWS Publishing Company, International Thomson Publishing Inc, Boston, 1996.

[27] H. A.van der Vorst, "Iterative Krylov Methods for Large Linear Systems," Cambridge University Press, Cambridge, United Kingdom, 2003.

[28] M. Benzi, G. H. Golub.,and J. Liesen, "Numerical solution of saddle point problems," Acta Numerica, vol.14, pp. 1–137, 2005.

[29] L. Bergamaschi and M. Ángeles, "Parallel Acceleration Krylov Solvers by Factrized Approximate Inverse Preconditioners," Lecture Notes in Computer Science, vol. 3402, pp. 623–636, 2005.

[30] H. Dag, "An Approximate Inverse Preconditioner and its implementation for

Conjugate Gradient method," Parallel Computing, vol. 33, pp.83-91, 2007.

[31] C. Wu,. L. Lai,. C. Yang and P. Chiu, "Using hybrid MPI and Open MP programming to optimize communication in Parallel loop Self-scheduling scheme for multicore PC clusters, " The Journal of Supercomputing (online),2009

[32] H. van der Vorst, " Efficient and iterative reliable methods for linear systems," Journal of Computational and Applied mathematics, vol. 149, pp.251 International Journal of Numerical Methods in Fluids, vol.24,pp. 671–691,1997

[33] J. D. Anderson, "Computational Fluid Dynamics (The basics with Applications), McGraw-Hill Book Co. Singapore.1995.

[34] E. Erturk , " Discussion on Driven Cavity Flow," International Journal of Numerical Methods in Fluids, vol.60(3), pp.275–295,2008.

[35] J. Hua,. and J. Lou, "Numerical Simulation of Bubble Rising in Viscous Liquid," Journal of Computational Physics," vol.222, pp.769–795,2007.

[36] M. Ketabdari,. M. Nobari,. And M. Larmaei, "Simulation of Waves Group Propagation breaking in Coastal zone using Navier–Stokes solver with an Improved VOF Free Surface Treatment," Applied Ocean Research, vol.30, pp. 130–143, 2008

[37] M. Rudman, "Volume-Tracking Methods for Interfacial Flow Calculations," International Journal of Numerical Methods in Fluids, vol.24, pp. 671–691,1997

[38] G. W. Cowles, "Parallelization of the FVCOM Coastal Ocean Model," International Journal of High Performance Computing Applications, vol. 22(2), pp.177–193, 2008.

[39] Y.Shang, "A Distributed Memory Parallel Gauss-Seidel Algorithm for Linear Algebraic Systems," vol. 57(8), pp. 1369–1376, 2009.

[40] S. Zaleski, "Fully Multidimensional Flux-Corrected Transport Algorithms for Fluids," Journal of Computational Fluids, vol.31, pp. 335–362, 1979.

[41] M. Beltràn and A. Guzmàn, "How to balance the load in Heterogeneous clusters," International Journal of High Performance Computing Applications, vol. 23(1), pp. 99–118,2009.

[42] K. Li, "Fast and Scalable Matrix computation s on Distributed memory Systems," In 19th IEEE International Parallel and Distributed Processing Symposium, 2005.

[43] D. J. Mavriplis,. M. J. Aftosmis,. And M. Berger, "High Resolution Aerospace Applications Using the NASA Columbia Supercomputer," International Journal of High Performance Computing Applications, vol. 21(1), pp. 106–126, 2007.

[44] D. Mezher and B.Philippe, "Parallel Computation of pseudospectra of large sparse matrices," Parallel Computing, vol.28, pp.199–221, 2002.

[45] J. Straubhaar, "Parallel Preconditioners for the Congugate Gradient Algorithm using Gram-Schmidt and least square methods," Parallel Computing,vol.34(10), pp. 551–569, 2008.

[46] D. Hecquet,. H. Ruskin, and M. Crane, "Optimisation and parallelisation strategies for Monte Carlo Simulation of HIV infection, "Computers in Biology in Medicine, vol.37 (5), pp. 691–699, 2007

[47] W.Tao,. D. Cai,. X, Yan,. And B, Lembege, "Scalability analysis of parallel Particle in Cell codes on computational grids," Computer Physics Communication, vol.179, pp. 855–864, 2008.

[48] O. Beaumont., A. Legrand., F. Rastello, and Y. Robert, "Static LU Decomposition on Heterogeneous Platforms," International Journal of High Performance of Computing Applications, vol.15(3), pp. 310–323, 2001.

[49] L .Dalcin, R. Paz, M. Storti, and J. D'Elia, "MPY for Python: Performance improvements and MPI-2 extensions," Journal of Parallel and Distributed Computing, vol. 68, pp. 655–662, 2008.

[50] T. T. Le and J. Rajeb, "A detailed MPI communication model for distributed system," Future Generation Computer System, vol.22, pp. 269–278, 2006.

[51] M. Quinn, "Parallel Programming in C and MPI and open MP," Dubuque, Iowa McGraw-Hill, 2004.

[52] R. Duncan, "A Survey of Parallel Computer Architectures," Computer, vol. 23(2), pp. 5–16, 1990,

****